Sistema de Telemonitoramento de Baixo Custo Usando IoT

Bianca Lisle
Departamento de Eletrônica e Sistemas
Universidade Federal de Pernambuco
blvaguiar@gmail.com

João Marcelo Teixeira
Departamento de Eletrônica e Sistemas
Universidade Federal de Pernambuco
jmxnt@cin.ufpe.br

***Resumo*—O avanço da tecnologia proporcionou o surgimento do conceito de Internet das Coisas, que está cada vez mais presente no cotidiano de pessoas, empresas e indústrias. Produtos conectados oferecem uma experiência diferenciada ao usuário, que pode interagir e receber informações e funcionalidades adaptadas para seu consumo e padrão. Este trabalho propõe um sistema de monitoramento por vídeo utilizando Internet das Coisas, com foco no monitoramento de crianças para detecção de movimento. O projeto utiliza uma Raspberry Pi Zero W, uma câmera e a biblioteca OpenCV para monitorar, processar e enviar o vídeo e suas informações para outros dispositivos conectados na mesma rede WiFi. Além disso, o projeto também visa trazer uma alternativa simples e barata para que pais possam monitorar crianças através de qualquer aparelho conectado.**

***Index Terms*—Telemonitoramento, IoT, Raspberry Pi**

***Abstract*—The growth of the Internet of Things concept is straightly connected to the current technological advances. IoT is increasingly present in people's lives, companies, and industry. Connected products provide a differentiated experience to the user, who can interact and receive customer-specifc information and functionalities. This work aims to propose a video monitoring system applying the Internet of Things, with a focus on child monitoring and movement detection. The project utilizes a Raspberry Pi Zero W, a camera module and the OpenCV library to monitor, process and send video and it's information to other devices connected to the same WiFI network. Also, the project intends to provide a simple and cheaper alternative so that parents can monitor children from any connected device.**

***Keywords*—Telemonitoring, IoT, Raspberry Pi**

I. Introdução

O advento da Internet das Coisas (IoT) nos últimos anos ocasionou o surgimento de dispositivos inteligentes e conectados, com capacidade de compartilhar diversas informações em tempo real com outros dispositivos conectados através da mesma tecnologia. Novas demandas e produtos estão surgindo no mercado [1], outros problemas podem ser mitigados e as soluções propostas podem estar diretamente ligadas à IoT.

Um caso de uso de como a IoT pode modificar as mais diversas experiências de usuário é a utilização das fechaduras inteligentes. Uma fechadura inteligente avançada e já disponível no mercado [2] permite que o seu usuário tranque e destranque a casa remotamente, além de mitigar a necessidade de utilização de chaves, através da conexão da fechadura com a internet. Outras funcionalidades são oferecidas pela fechadura inteligente. Uma delas é a capacidade de acompanhar o histórico de aberturas da fechadura através de um aplicativo. A utilização por meio de comando de voz e integração de plataformas IoT são implementadas afim de construir o conceito de uma casa inteligente. Tendo em vista a utilização de dispositivos IoT para a resolução de novos problemas, uma solução a ser explorada é o monitoramento por vídeo, mais especificamente, de crianças. As atuais babás eletrônicas de preçco acessível, com um custo em torno de 100 reais, são compostas por dois dispositivos se comunicam através de radiofrequência, transmitindo apenas áudio.

As babás eletrônicas mais avançadas podem ser compostas por dois aparelhos que transmitem vídeo e áudio ou até mesmo utilizar um dispositivo que envia mensagens para um smartphone, o que torna seu valor reduzido, além de reduzir a quantidade de dispositivos necessários [3]. Porém, ambas as soluções apresentam um valor mais elevado em relação às babás eletrônicas mais simples(R$100), e tornam-se aplicações complexas de utilizar em termos de logística.

Este trabalho tem por objetivo obter uma solução viável em termos de custo e implementação utilizando o conceito de IoT. Ou seja, buscar tecnologias e dispositivos mais baratos e que possam se comunicar com um smartphone ou qualquer outro dispositivo conectado e tratar os dados obtidos para que possa ser oferecida uma experiência simples e funcionalidades como a detecção de movimento no vídeo e o acompanhamento remoto do mesmo. Além disso, deseja-se que o sistema elaborado seja o mais versátil possível, podendo ser utilizado pelas mais diversas plataformas disponíveis no mercado, desta forma é possível que, mesmo com o avanço da tecnologia e surgimento de novos aparatos tecnológicos, o sistema elaborado ainda seja útil e viável. Também é objetivo deste trabalho avaliar a facilidade para desenvolver soluções IoT que tragam valor ao usuário e que sejam de baixo custo.

O restante do artigo está estruturado conforme segue. A Seção 2 exibe os trabalhos existentes relacionados ao sistema proposto. Em seguida, a Seção 3 descreve a arquitetura do sistema proposto, além de retratar os procedimentos adotados para a execução deste trabalho. Na Seção 4, avalia-se o funcionamento do sistema para validação. Por fim, a Seção 5 expõe as considerações finais referentes ao trabalho desenvolvido, além de propor melhorias para o mesmo.

II. Trabalhos relacionados

A análise dos trabalhos existentes e relacionados ao tema foco do projeto proposto é fundamental para a construção de um trabalho útil e que agrega valor ao estado da arte do tema abordado. Assim, um estudo sobre estes trabalhos foi feito, a fim de utilizá-los como referência técnica e também de ideação do trabalho proposto. O artigo "A Prototype of Child Monitoring System Using Motion and Authentication with Raspberry Pi", escrito em 2016 [4], aborda um protótipo para monitoramento de crianças com uma Raspberry Pi B+ e uma câmera USB. O trabalho também propõe detecção de movimento, utilzando a biblioteca motion [5] para detectar mudanças. Além disto, o paper propõe o acesso remoto do monitoramento com uma funcionalidade de autentica ção. Em 2017, Kamran Ayub desenvolveu um sistema de monitoramento open source para bebês em seu blog em forma de tutorial em seu blog pessoal [6]. O projeto era composto por uma Raspberry Pi 3, uma câmera para Raspberry e uma Lâmpada LED USB. Assim, o vídeo é capturado através da câmera e o vídeo é servido em uma URL específica que pode ser acessada através do player VLC. A lâmpada é utilizada para o modo noturno de monitoramento. Também é uma funcionalidade do projeto a exibição do vídeo em uma página web local ou na internet. No trabalho proposto por Kamran nenhum processamento de imagem é executado, e não é oferecida nenhuma interface simples de configuração do módulo de monitoramento. No ano de 2018, Jonathan Ervine fez uma publicação explicando como construir um monitor para crianças com uma Raspberry Pi. O sistema é composto por uma Raspberry Pi 3 e um sensor de temperatura e umidade do ambiente. Utilizando Python e um script em Perl, o mesmo contruiu o sistema que envia os dados para um banco de dados MySQL e em seguida os dados são exibidos em uma página web utilizando Google Charts.

Conforme os projetos apresentados, o trabalho proposto visa utilizar a Raspberry Pi Zero W, com menor custo, além de oferecer a funcionalidade de detecção de movimento utilizando a biblioteca OpenCV, que possui diversos algoritmos de processamentod de imagem.

III. Metodologia

Esse capítulo abordará a metodologia da implementação desse traballho. Devido à característica experimental deste projeto, durante o processo de ideação e desenvolvimento diversas questões foram levantadas e alguns requisitos foram ajustados de acordo com os objetivos e viabilidade do projeto. A seção de arquitetura do sistema aborda a metodologia e os questionamentos levantados para tomar as decisões de desenvolvimento. As demais seções abordarão de maneira mais detalhada a elaboração deste trabalho.

A. Arquitetura do Sistema

O objetivo deste projeto foi criar um sistema de monitoramento simples, conectado e versátil. A primeira questão a ser discutida foi qual seria a plataforma utilizada para interface com o usuário. Atualmente, como explorado anteriormente, poderiam ser criados aplicativos iOS e Android para smartphones e tablets. Entretanto, uma vez que deseja-se que o dispositivo seja compatível com a maior quantidade possível de aparelhos conectáveis, decidiu-se utilizar uma página web como interface com o usuário no lugar da criação de aplicativos. Desta forma, qualquer dispositivo compatível com o protocolo HTTP pode obter informações sobre o sistema de monitoramento. Em seguida, a próxima questão foi sobre como seria feito o processo de estabelecimento de uma conexão entre smartphone e sistema de monitoramento. Além disto, como configurar o sistema de monitoramento para estabelecer uma conexão com o roteador do ambiente que será monitorado? Como trata-se de um dispositivo embarcado limitado sem display e sem teclados, foi preciso pensar em protocolos ou tecnologias disponíveis para fazer esta conexão.

Dentre as alternativas levantadas, as que se pode destacar foram: Bluetooth e Wi-Fi. Ambas destacaram-se pois são tecnologias comumente presentes em dispositivos IoT. O Bluetooth foi considerado uma importante opção devido a sua crescente utiliza ção em aplicações IoT na indústria [7]. é um protocolo simples e pode ter menor consumo de energia se a sua versão de baixa energia for utilizada. Além disso, a Raspberry Pi Zero W já possui comunicação bluetooth embutida, descartando a necessidade de comprar itens adicionais. Entretanto, para que tal tecnologia pudesse ser utilizada, seria necessário o desenvolvimento de aplicativos específicos para tal comunicação. Como um dos objetivos é a versatilidade do sistema, esta opção foi descartada, pois seria necessário criar uma aplicação para cada sistema operacional.

A alternativa do Wi-Fi tornou-se mais interessante pois diferente do Bluetooth, notou-se que existia uma alternativa que descarta a necessidade da criação de aplicativos. Utilizando essa tecnologia e com o sistema de monitoramento configurando um ponto de acesso Hotspot, foi possível estabelecer um canal entre os dispositivos clientes e o sistema de monitoramento. Ao conectar o Hotspot uma página web é criada, possibilitando a configuração da rede local. A Figura 1 esboça o sistema desenvolvido. O sistema de monitoramento é composto por uma Raspberry Pi conectada a uma rede Wi-Fi e a uma câmera de vídeo. As plataformas que são clientes deste sistema, que funciona como um servidor, conectam-se também à mesma rede Wi-Fi.

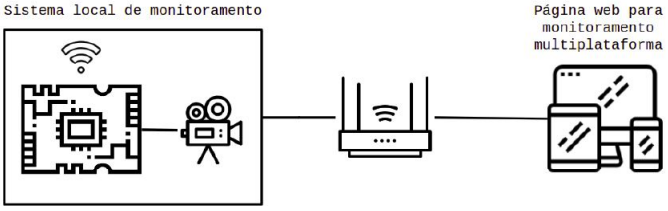

Figura 1. Arquitetura do sistema proposto.

Ou seja, inicialmente provê-se um ponto de acesso para que o usuário, em seu primeiro contato com o sistema, consiga configurar a rede local do espaço a ser monitorado. Em

seguida, a Raspberry conecta-se à rede informada e redireciona o usuário para a página de monitoramento, onde o vídeo processado é exibido.

B. Ambiente da Raspberry

Diversas operações e instalações foram executadas para o desenvolvimento do sistema. Essa seção aborda os principais tópicos para a garantia das principais funcionalidades do sistema.

1) Configuração do Hotspot: A configuração do Hotspot foi feita através de um script bash, que possibilita a execução de diversos comandos na linha de comando, inclusive de maneira condicional. O script foi disponibilizado em [BLIND], que é o repositório com todo o código deste trabalho. Através do código, os módulos DNSMasq e HostAPD foram instalados na Raspberry para possibilitar a configuração do ponto de acesso [8]. Em seguida, o endereço de IP estático foi configurado e o serviço DHCPCD foi reiniciado. Em seguida, o endereço IP da Raspberry foi configurado.

Para fazer o sistema funcionar como um ponto de acesso utiliza-se o programa hostapd. O programa é projetado para rodar em plano de fundo no sistema como um componente que controla pontos de acesso e autenticação. Para isso ele implementa o controle de ponto de acesso do IEE 802.11 (rede Wi-Fi) [9]. Este comando especifica quais os números de IP disponíveis na rede, com um intervalo de 24 horas. Já a configuração para a rede de acesso gerada foi especificada no arquivo hostapd.conf.

2) Instalação OpenCV: Este trabalho utiliza a biblioteca OpenCV para a detecção de movimento no vídeo transmitido. Essa biblioteca foi escolhida devido a sua vasta utilização e versatilidade, e portanto possibilidade de utilizar outros algoritmos de processamento de imagem e vídeo no futuro. Para instalar o OpenCV na Raspberry Pi Zero W é necessário primeiro instalar todas as dependências (que são numerosas) e incluem pacotes de interface de entrada e saída para vídeo, ferramentas de desenvolvimento, pacotes de otimização e a versão de Python de sua escolha. Vale ressaltar que o processo de compilação da biblioteca não foi simples, conforme abordado na seção de dificuldades. A maioria dos projetos e tutoriais existentes com OpenCV utilizam a Raspberry Pi 3. Em seguida, o código-fonte do OpenCV foi baixado do seu repositório oficial do Github e compilado usando as ags recomendadas como em [10] e desabilitando a pré-compilação de headers, pois se habilitada, o processo de compilação falha.

C. Hardware

A configuração de hardware do sistema proposto utiliza a Raspbery Pi Zero W, que já possui módulo Wi-Fi enbutido, utilizando o sistema operacional Raspbian Stretch. O Raspbian é um sistema operacional baseado em Debian, grátis e otimizado para o hardware da Raspbery Pi [11]. Uma câmera para Rasperry Pi foi conectada ao dispositivo embarcado para a aquisição dos vídeos.

como pretendido, é menor comparado aos valores das soluções existentes no mercado.

D. Servidor

Para a construção do servidor, o framework Flask foi utilizado. As seções a seguir descrevem os dois estados do servidor, modo de configuração e modo de monitoramento.

1) Modo de configuração: O modo de configuração é o modo padrão em que o sistema se encontra ao ser ligado pela primeira vez. Neste modo a Raspberry Pi é um ponto de acesso de rede e disponibiliza uma página web que requisita do usuário o SSID e a senha da rede Wi-Fi à qual o sistema deve inicialmente se conectar.

Após inserir os dados necessários o usuário deve aguardar enquanto o sistema encerra o ponto de acesso e conecta-se a rede local desejada. O servidor então salva as informações da rede Wi-Fi e reinicia a interface responsável por conectar-se a uma rede local Wi-Fi. Ao fazer isso, o sistema troca de estado para o modo de monitoramento e passa a disponibilizar o vídeo que está sendo gravado em tempo real e exibe se movimentos estão sendo detectados ou não.

E. Modo de Monitoramento

O modo de monitoramento é acionado após a configuração do Hotspot. Neste modo, o servidor é redirecionado pra a rota de Streaming. Neste modo, o sistema processa e disponibiliza o vídeo para a página web de monitoramento. O processamento é feito através da biblioteca OpenCV utilizando a técnica de subtração fundo [12]. O método de detecção foi baseado na segmentação entre plano de fundo e primeiro plano [13]. Além disto, também assume-se que o plano de fundo do ambiente monitorado é, na maior parte do tempo, estático e sem mudanças bruscas durante uma quantidade consider ável de frames. Assim, assume-se que mudanças abruptas representam um movimento no vídeo. Diante desta suposição, espera-se que a câmera do sistema de monitoramento seja posicionada fixamente e sob luz controlada. Assim, o código de monitoramento inicializa uma stream de vídeo e processa os frames recebidos. O tratamento dos frames é feito da seguinte maneira: são redimensionados, convertidos para escala de cinza e por fim são embaçados, conforme mostrado a seguir.

```
frame = cv2.resize(frame, (500,500),
    cv2.INTER_AREA)
gray = cv2.cvtColor(frame,
    cv2.COLOR_BGR2GRAY)
gray = cv2.GaussianBlur(gray, (21, 21), 0)
```

Em seguida, o frame atual é comparado ao frame capturado anteriormente e então é calculada a difrença absuluta entre ambos. A diferença calulada é limitada para que apenas mudanças significantes sejam classificadas como movimento.

```
frameDelta =
    cv2.absdiff(firstFrame, gray)
thresh =
    cv2.threshold(frameDelta, 25, 255,
        cv2.THRESH_BINARY)[1]
```

Por fim, utiliza-se um código de detecção de contorno para que, caso a área do contorno seja maior que o limite estabelecido, uma caixa retangular seja desenhada na imagem.

IV. Experimentos e análise dos resultados

Para validar o projeto proposto, experimentos foram feitos a fim de verificar as se as funcionalidades propostas foram implementadas corretamente. Assim, esta seção detalhará os experimentos realizados e os resultados obtidos.

A. Funcionamento Hotspot

Para verificar o funcionamento do hotspot, o experimento realizado consistiu em conectar-se a rede local roteada e acessar o endereço de IP estático configurado anteriormente.

O experimento mostrou que o smartphone conectou-se com sucesso ao hostpot e obteve acesso à página de configuração através IP do servidor (192.168.4.1) que foi configurado estaticamente, como mostrado anteriormente. Desta forma, pode-se dizer que a configuração do hotspot funcionou de acordo como esperado.

B. Detecção de movimento

O experimento de detecção de movimento foi feito da seguinte forma: como não existiu a possibilidade de testar com crianças, foi necessário encontrar um meio de simular movimento e validar que o mesmo foi detectado corretamente. Para isto, foi escolhido um ambiente calmo e com pouca variação de movimento, uma sala. Além disto, como dito anteriormente, é necessário garantir que o ambiente é estável, portanto, a Raspberry Pi precisou de uma base fixa.

Para simular o movimento, foi utilizado um ventilador com rotação um eletrodom éstico presente em muitas residências e que executa um movimento notável e contínuo. A Figura 2 mostra o sistema no navegador Google Chrome de um computador com o sistema operacional Ubuntu de monitoramento com o ventilador parado.

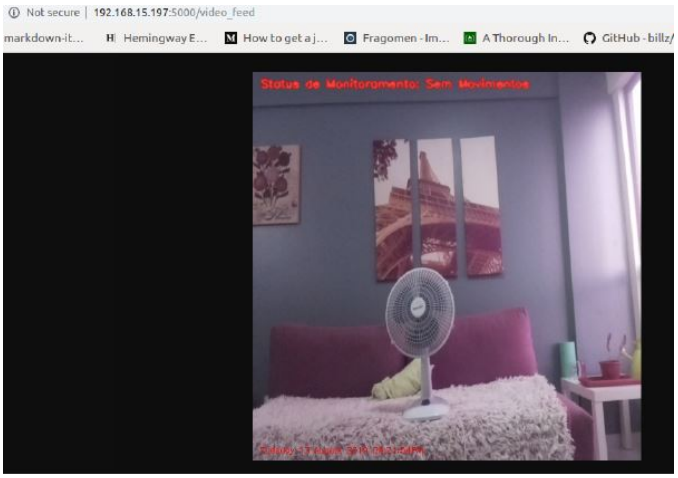

Figura 2. Sistema de monitoramento com o ventilador desligado.

Em seguida, o ventilador foi ligado com rotação habilitada e dessa forma, notou-se que o sistema foi capaz de detectar seu movimento, como mostrado a seguir na Figura 3.

Além da capacidade de detectar movimento, também foi foco do experimento estimar a latência para a detecção do movimento. Foram medidas as taxas de quadros por segundo de acordo com as configurações listadas na Tabela I.

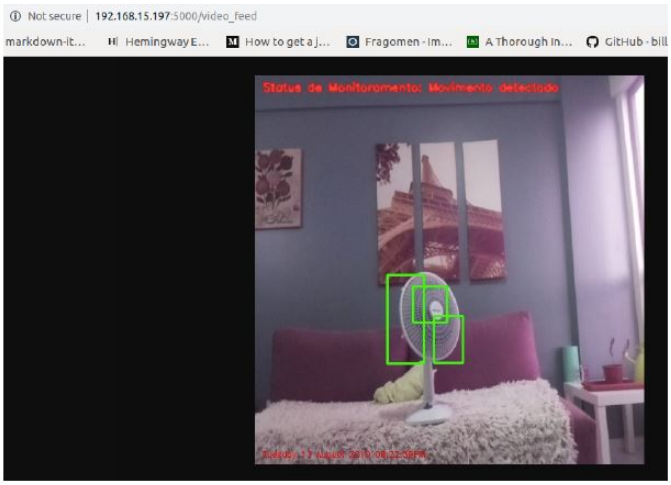

Figura 3. Sistema de monitoramento com o ventilador em movimento.

Tabela I
Tabela 2: Taxas de quadros por segundo do sistema.

Configuração	Taxa
Sem processamento	4,87
Com processamento e sem movimento	2,068
Com processamento e movimento	2,11

É possível notar que o impacto do processamento da imagem reduz a taxe de frames por um valor maior que a metade da taxa original. Ainda com uma taxa baixa, foi possível verificar movimentos, como mostrado anteriormente. Entretanto é importante ressaltar que mecanismos para o aumento da taxa de quadros por segundo devem ser estudados como uma possível melhoria. Também foi estimado o tempo para detecção de movimento utilizando um cronometro e verificando o time stamp do sistema. O tempo obtido foi de aproximadamente 1,5 segundos. Assim, foi possível avaliar que a detecção de movimento funcionou de modo esperado, cumprindo este objetivo.

C. Monitoramento multiplataforma

Como visto nas seções anteriores, o sistema funcionou de acordo com o esperado para a plataforma web de um computador e para o navegador Safari do iOS. Por fim, foi preciso testar o sistema em um dispositivo Android com o navegador Mozilla Firefox. O experimento foi similar ao teste executado na seção de monitoramento, agora com o aparelho citado.

V. Conclusão

No decorrer deste trabalho, diversos temas foram explorados, conceitos e tecnologias abordados durante a graduação foram aprofundados e postos em prática. Desta forma, o sistema mostrou-se funcional de acordo com os resultados obtidos. Além disto, muitas dificuldades foram encontradas

e solucionadas através de um vasto processo de pesquisa, pois apesar da bibliografia e tutoriais existentes atualmente, outros erros foram apresentados e solucionados.

Diante dos resultados obtidos, das dificuldades apresentadas e dos trabalhos relacionados, foi possível analisar e propor possíveis melhorias para a garantia de um sistema com mais funcionalidades e que forneça uma melhor experiência para o usuário. Além disto, muitas decisões de implementação foram tomadas de maneira a garantir a possibilidade de implementação das melhorias propostas.

Desta forma, um aperfeiçoamento sugerido para o sistema seria a transmissão de áudio e vídeo, garantindo que o monitoramento possa ser feito não só através das imagens mas também de áudio. Além disto, seria interessante também aplicar técnicas de processamento digital de sinais para a identificação de sons característicos, como o choro de uma criança, um barulho muito alto, entre outros. Outra possível melhoria é a utilização de outros algoritmos de processamento de imagem e vídeo do OpenCV para proporcionar ao usuário mais funcionalidades de monitoramento.

Para isto, também seria necessário encontrar métodos para que a taxa de quadros por segundo seja maior, como citado anteriormente. Seria também interessante adicionar a funcionalidade de monitoramento não apenas local (na mesma rede Wi-Fi) mas também à distância, porém, juntamente com essa alteração, é importante adicionar mecanismos de segurança para a transmissão do vídeo.

Por fim, outra modificação sugerida é a remoção do adaptador Wi-Fi extra. Uma vez que o sistema consiga redirecionar corretamente o usuário do Hotspot para o endereço de IP adquirido após conexão com a rede local, não há necessidade da utilização do adaptador. Isto pode ser feito através do envio de um e-mail com o novo link para monitoramento, por exemplo.

Diante das dificuldades e melhorias propostas, pode-se afirmar que o objetivo deste trabalho foi satisfeito, uma vez que foi possível construir um sistema simples e eficaz de monitoramento com possibilidade de diversos aprimoramentos e o mapeamento das dificuldades apresentadas.

REFERÊNCIAS

[1] "IoT Devices List | Smart Home Devices | Top Internet of Things Devices." [Online]. Available: http://iotlineup.com/

[2] "August Smart Lock Pro | The Ultimate Smart Lock for Your Smart Home." [Online]. Available: https://august.com/products/august-smart-lock-pro-connect

[3] "ITU-T Recommendation database." [Online]. Available: https://www.itu.int/ITU-T/recommendations/rec.aspx?rec=11559lang=en

[4] O. Permatasari, S. U. Masruroh, and Arini, "A prototype of child monitoring system using motion and authentication with Raspberry Pi," in *2016 4th International Conference on Cyber and IT Service Management*, Apr. 2016, pp. 1–6.

[5] "Motion - Open source security camera software." [Online]. Available: https://motion-project.github.io/

[6] K. Ayub, "Kamranicus - Building a Raspberry Pi 3 Baby Monitor." [Online]. Available: https://kamranicus.com/guides/raspberry-pi-3-baby-monitor

[7] "How Bluetooth Enhances IoT Architecture To Next Level | Enabler." [Online]. Available: https://www.cyient.com/enabler/market/bluetooth-a-smart-communication-protocol-for-the-internet-of-things.php

[8] "Setting up a Raspberry Pi as a Wireless Access Point - Raspberry Pi Documentation." [Online]. Available: https://www.raspberrypi.org/documentation/configuration/wireless/access-point.md

[9] "Updating hostapd fails… - Raspberry Pi Forums." [Online]. Available: https://www.raspberrypi.org/forums/viewtopic.php?t=235145

[10] A. Rosebrock, "Installing OpenCV on your Raspberry Pi Zero," Dec. 2015. [Online]. Available: https://www.pyimagesearch.com/2015/12/14/installing-opencv-on-your-raspberry-pi-zero/

[11] "Raspbian - Raspberry Pi Documentation." [Online]. Available: https://www.raspberrypi.org/documentation/raspbian/

[12] "OpenCV: Background Subtraction." [Online]. Available: https://docs.opencv.org/3.2.0/db/d5c/tutorial_py_bg_subtraction.html

[13] A. Rosebrock, "Basic motion detection and tracking with Python and OpenCV," May 2015. [Online]. Available: https://www.pyimagesearch.com/2015/05/25/basic-motion-detection-and-tracking-with-python-and-opencv/